

**VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky**

**Návrh řešení propojení informačního systému Vario 12 s
internetovým obchodem**

**The Solution of an Integration Between VARIO 12 ERP and E-
shop**

2013

Michal Sikora

Zadání bakalářské práce

Student:

Michal Sikora

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R025 Informatika a výpočetní technika

Téma:

Návrh řešení propojení informačního systému Vario 12 s internetovým obchodem

The Solution of an Integration Between VARIO 12 ERP and E-shop

Zásady pro vypracování:

Cílem bakalářské práce je návrh a popis řešení datového propojení mezi systémem pro internetové obchodování a ekonomickým informačním systémem Vario 12.

1. Popis problému, nástin možných řešení.
2. Definice datového rozhraní pro přenos dat z ERP.
3. Příprava jednoduché webové aplikace (eshopu) se základními funkcemi.
4. Návrh a realizace datového propojení IS a eShopu - přenos dat do eShopu (zboží, kategorie, zařazení zboží do kategorie, ceníky, zákazníci, přehledy vystavených objednávek).
5. Návrh a realizace datového propojení IS a eShopu - přenos dat do IS (registrace nových zákazníků, objednávky).
6. Dokumentace.

Seznam doporučené odborné literatury:

Podle pokynů vedoucího bakalářské práce.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Jaromír ChochoLATý**

Datum zadání: 18.11.2011

Datum odevzdání: 07.05.2013



doc. Dr. Ing. Eduard Sojka
vedoucí katedry




prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení studenta

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

Dne: 01.05.2013


.....
podpis studenta

Poděkování

Rád bych poděkoval Ing. Jaromíru Chocholatému za odbornou pomoc a konzultaci při vytváření této diplomové práce.

Abstrakt

Tato práce se zabývá popisem a návrhem výměny dat mezi ekonomickým softwarem Altus Vario a internetovým obchodem. První je vysvětlen samotný ERP systémem Altus Vario, co vlastně dělá a k čemu je určen. Práce dále popisuje problematiku výměny dat, návrhy propojení pomocí existujících řešení, nebo vlastních a výběr jednoho z nich. Nakonec je řešení otestováno pomocí jednoduché webové aplikace.

Klíčová slova

Ekonomický program, ERP, internetový obchod, databáze, MS SQL

Abstract

This paper describes a proposal for data exchange between economic Altus Vario software and online store. The first explains itself ERP system Altus Vario what he was doing and what it is intended. The work also describes the problem of data exchange, link suggestions by existing solutions, or a custom selection of one of them. Finally, the solution is tested by using a simple web application.

Key words

Economic program, ERP, internet trade, database, MS SQL

Seznam použitých zkratek

Zkratka	Anglický význam	Český význam
ERP	Enterprise Resource Planning	Plánování podnikových prostředků
IIS	Internet Information Services	Internetová informační služba
MS SQL	Microsoft Structured Query Language	Jazyk pro práci s relačními databázemi
XML	Extensible Markup Language	Rozšiřitelný značkovací jazyk
XSD	XML Schema Definition	XML schéma
OLE DB	Object Linking and Embedding	API pro přístup k datům
HTML	HyperText Markup Language	Značkovací jazyk pro hypertext
FTP	File Transfer Protocol	Protokol pro přenos souborů

Obsah

1	Úvod	1
2	Popis problému, nástin možných řešení.....	2
2.1	Specifikace	2
2.2	Popis problematiky	3
2.2.1	Propojení Altus Varia s internetovým obchodem.....	4
2.2.2	Altus Vario a internetový obchod.....	4
2.2.3	Zdroje dat.....	4
2.2.4	Databáze Altus Varia.....	4
2.2.5	Databáze internetového obchodu.....	5
2.2.6	Výměna dat mezi databázi (jednosměrná, obousměrná)	5
2.2.7	Program pro výměnu dat	5
2.2.8	Obchodní procesy	6
2.2.9	Hotová řešení.....	7
2.2.10	Založení knih	7
2.2.11	Pořadí importovaných souboru, dat.....	8
2.3	Nástin možných řešení	8
2.3.1	Výměna dat přes XML rozhraní	8
2.3.2	Altus Vario Integration Services	13
2.3.3	Přímé propojení databází	16
2.4	Výběr vhodného řešení.....	19
3	Definice datového rozhraní pro přenos dat z ERP.	20
4	Návrh a realizace.....	22
4.1	Použitý software	22
4.2	Uložené procedury	22
4.3	Výměna dat, postupy a pravidla	23
4.4	Aktualizace dat, knihy agend datum aktualizace.....	23
4.5	Postup při přenášení dat.....	23
4.6	Směry toku dat.....	23
4.7	Zajištění obnovy dat v určitém intervalu.....	24
4.8	Co když server Varia v době odeslání objednávky a registrace nebude dostupný? ..	25

4.9	Online pohledy	25
5	Příprava jednoduché webové aplikace se základními funkcemi	26
5.1	Použitý software a technologie.....	26
5.2	Popis webové aplikace	26
6	Závěr	27
	Použitá literatura.....	28
	Seznam příloh	XXIX

1 Úvod

V dnešní době se velmi rychle rozšířilo obchodování přes internet. Zákazník v pohodlí svého domova, kdekoliv, odkudkoliv má přístup k internetu a nějaké zařízení (mobil, notebook atd.) na němž může provozovat webový prohlížeč si může objednat jakékoliv produkty a nechat si je dovézt až domů. To vše aniž by musel pro to podniknout dlouhou cestu a nakonec by stejně jen dozvěděl, že výrobek není na skladě. Ano, až tak snadné to je.

Existuje mnoho různých aplikací, softwaru, systému zefektivňující toto podnikání. Jedná se zejména o e-shop, internetový obchod je speciální webová aplikace sloužící k provádění obchodních transakcí na internetu. A dále jsou tu desktopové aplikace jako je Altus Vario (spadá do kategorie ERP systému) užívané například při naskladňování zboží, vystavování faktur, vedení záznamů o zaměstnancích.

E-shop běžně obsahuje služby, zboží s možností si je objednat, stará se o příjem objednávek od zákazníků, zprostředkování plateb a poskytování dalších informací o výrobcích či řešení reklamací. Poté co si zákazník vybere produkt, dochází k nákupnímu procesu. Produkt přidá do virtuálního nákupního košíku, místo kde se shromažďují všechny přidávané produkty. Pokud je zákazník spokojen, má vybráno všechno co chtěl a přidáno do košíku, může pokračovat k dalšímu kroku. Vyplní své osobní údaje a místo doručení, pokud není nutná registrace, dále si vybere způsob doručení a platby. Vše potvrdí a pak už jen vyčkává na objednané zboží.

Abych to všechno shrnul a popsal hlavní náplň zadání. Cílem této práce je navrhnout propojení, konkrétně datové propojení Altus Varia s internetovým obchodem, tak aby přenos dat byl maximálně validní a data obou databází byla po každé změně v korektním stavu. Altus Vario při přidávání nových položek agend, hlavně se jedná o položky agend katalogu, ceníku, skladu, adresáře aj., si ukládá svá data do databáze buď na lokální disk, nebo na vzdálený server. Internetový obchod bude mít možnost, jak už většina populace vědí, se registrovat a hlavně vytvářet objednávky za účelem nákupu zboží a pro administrátory obchodu bude nastavení navíc obsahovat jednoduché funkce pro obnovu dat mezi oběma systémy. Databáze internetového obchodu je většinou na stejném serveru neboli také hostingu, spolu s internetovým obchodem, který v mém případě poběží na webovém serveru IIS.

2 Popis problému, nástin možných řešení

Úvodem téhle kapitoly bych chtěl popsat a přiblížit slovně i za pomoci ilustrací popis problému, nástin možných řešení a s tím související jednoduchý popis použitých technologií při kompletování systému tak, aby dalo všem co jak nejjasnější smysl.

Altus Vario jak už jsem zmínil v úvodní kapitole je ERP (Enterprise Resource Planning) ekonomický informační software sloužící ke zpracování velkého množství procesů. Jedná se například o výrobu, distribuci, logistiku, správu majetku, prodej, fakturaci, účetnictví nebo též tzv. firemních agend jako je například vedení obchodních, účetních a personálních agend. Agendy jsou rozděleny do jednotlivých specializovaných modulů. Tyto moduly lze jednotlivě používat, dokoupit, v závislosti na tom, jaké firma potřebuje pro svoji činnost. Jak bych já řekl, každá firma si může svůj ERP systém ušít na míru a neplatit zbytečně za služby, které nepotřebuje. Záleží, co potřebuje. S pomocí softwaru Altus Vario lze řešit tyto obchodní aktivity:

- nákup a prodej zboží a služeb (sklady, zakázky, objednávky, fakturace, pokladna)
- vedení účetnictví, personalistiky a mezd, evidence majetku a další
- práce s klienty (adresář, deník aktivit)
- řízení vztahů se zákazníky (CRM)

2.1 Specifikace

Firemní systém Altus Vario pro řízení podniků je vyvíjený pro malé a střední firmy. Jeho vysoký výkon totiž bez problému pojme 100 uživatelů souběžně. I když tak vysokou kapacitu některé firmy nevyužijí, vyplatí se mít vždy nějakou rezervu a pro malé firmy jsou, které tak velký potenciál nepotřebují, jsou kategorie cenově zvýhodněny.

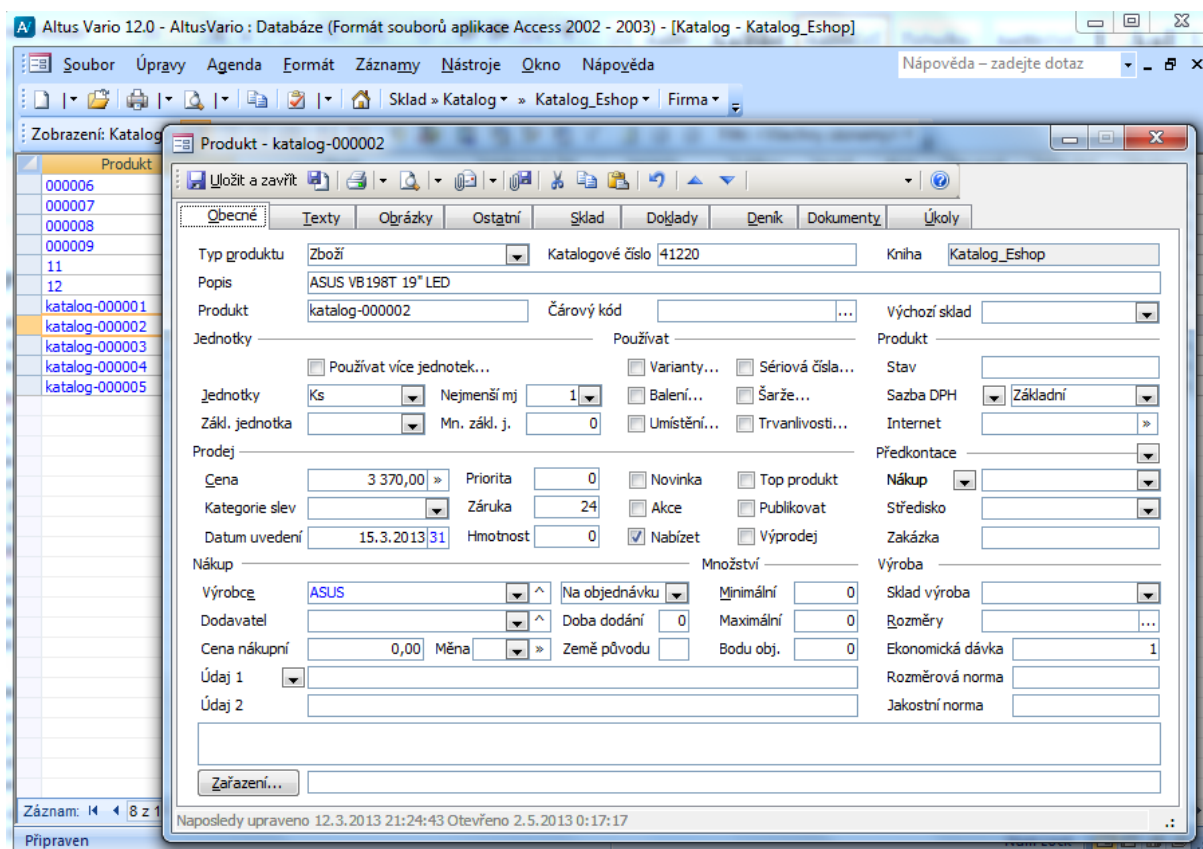
Mezi jeho další vlastnosti patří spolehlivost, především kvůli pravidelným aktualizacím včetně složité legislativy ČR/EU a také vlastní certifikát ISO 9001 v oblasti vývoje a implementace software.

Na první pohled se Altus Vario podobá standardně používaným kancelářským aplikacím Microsoft Office, což je jen další z výhod, jelikož uživatelé můžou pracovat v již známém a přívětivém prostředí.

Asi nejsilnější zbraní Varia lze považovat jeho přizpůsobení požadavkům klienta pomocí různých nástaveb a doplňků (modulů) nejen od výrobce, ale také od nezávislých dodavatelů, ze kterých si zákazník poskládá svůj ideální systém.

U firem, ve kterých jsou jednotlivé agendy rozděleny pro jednotlivé zaměstnance, nebo oddělení lze každý modul provozovat samostatně a zároveň je lze všechny nainstalovat jednotlivě na jediný počítač.

V závislosti na počtu uživatelů a počtu položek mohou být databáze na vzdáleném SQL serveru, síťovém souborovém serveru, nebo na lokálním disku bez použití sítě za předpokladu, že systém běží na jednom počítači. Informační systém Altus Varios díky těmto technologiím může být skvělým řešením firem a poboček různých velikostí. Za zmínku také stojí, že umožňuje výměnu dat mezi centrálou a pobočkami.



Obrázek 4 .1: Altus Vario – úprava položky v agendě Katalog.

2.2 Popis problematiky

Výměna dat informačního systému Altus Varia s internetovým obchodem a naopak by mělo být maximálně automatizováno a korektně ošetřeno aby koncový zákazník mňel co nejmenší námahu s obnovováním dat mezi oběma systémy a aby si datové typy tabulek obou databázi odpovídaly.

Dejme tomu například nově vzniklé objednávky, které vznikají na straně internetového obchodu. Aby je uživatel nemusel ručně přepisovat z obchodu do agendy, Zakázky systému Varia lze pár krát kliknout myší anebo pomoci nastaveného intervalu se všechny objednávky aktualizují během kratičké chvíle. Toho lze docílit mnoha způsoby a vždy po nějaké době s nově nabitými zkušenostmi a přibývajícimi technologiemi použitého programovacího jazyka lze tyto metody zjednodušovat, snižovat náklady na přenos dat a lépe systémy zabezpečovat.

2.2.1 Propojení Altus Varia s internetovým obchodem

Altus Vario a internetový obchod představuje dva odlišné systémy. Každý z nich má svoje datové úložiště pro ukládání dat a oba jsou na různých místech. Tím můžeme rozumět například různé místo umístění serveru, na kterém systém běží, nebo že oba systémy mohou být provozovány na různých softwarových platformách ať už je to programovací jazyk nebo operační systém.

V dalších slovech bych chtěl nastínit, jak propojit takový ERP systém s internetovým obchodem. Existuje mnoho řešení jak to udělat, ale ne všechny budou splňovat ideální řešení. Především jde hlavně o automatizaci a dynamičnost služby. Při návrhu řešení propojení se ukáže najevo, co jsem tím myslel. Dále bych se chtěl zaměřit na popis datových struktur pro výměnu dat mezi obchodem a Variem. Také existují hotová řešení, o kterých padne taky zmínka.

2.2.2 Altus Vario a internetový obchod

Altus Vario je firemní nástroj pro řízení obchodních agend, ale pouze některé z nich jsou použitelné pro obchodování v internetovém obchodě. Mezi ty užitečné patří tyto agendy:

- katalog zboží, ceníky a sklady,
- knihy došlých objednávek (ve Variu se nazývají zakázky),
- adresář (zákazníci)
- knihy prodejních dokladů (faktury a pokladní doklady, dále jen faktury).

Popis těchto agend je zaměřen na výměnu dat mezi Variem a internetovým obchodem. Výměnou dat je myšlena jakákoliv komunikační služba, která zajišťuje automatickou výměnu dat. Altus Vario nabízí přímo jednu takovou, konkrétně webovou službu s názvem Altus VarioIntegration Services, u které se nejedná o ruční export a import dat. Pro méně rozsáhlejší obchody lze použít průvodce exportem a importem dat, který nabízí několik definovaných formátů, mezi které patří například XML, XLS aj. nastavení pro export zboží a import objednávek.

2.2.3 Zdroje dat

Altus Vario a internetový obchod využívá dva zdroje dat, každý systém má jednu databázi dat. U Varia je dokonce volba mezi dvěma zdroji dat možná hned při instalaci systému. Buď je na výběr Microsoft Access, nebo MS SQL, a který si zvolíme, záleží jen na Nás, ale pro mé účely jsem si nakonec zvolil vytvoření datové struktury v MS SQL. A nakonec musí být vytvořená komunikace mezi oběma databázemi dat, třeba nějakou exe aplikaci, Windows službou, nebo se výměna dat může provádět pomocí funkcí na straně internetového obchodu.

2.2.4 Databáze Altus Varia

Tato databáze je umístěna ve firemní síti, odkud je povolen přístup do internetu. Přístup z internetu do sítě je striktně zakázán, zabezpečen pomocí firewallu. Pro zajištění spolupráce s obchodem a manipulace s daty databáze obchodu jsou dvě možnosti:

1. Veškerý přístup z internetu bude nadále zakázán a výměna dat mezi oběma databázemi realizována pomocí aplikace z vnitřní sítě firmy s Variem.
2. Druhou možností je povolení přístupu konkrétní aplikace nastavením firewallu a to pro konkrétní port, protokol a také za pomoci autorizace

První možnost se označuje jako jednosměrná komunikace a druhá je obousměrná komunikace. Databáze Varia určena pro výměnu dat s obchodem se označuje jako Dataxxxx, kde tvar xxxx je číslo, které se zadává při instalaci Varia. Defaultně je nastaveno na 0001, ale může to být jakékoliv čtyřmístné číslo.

2.2.5 Databáze internetového obchodu

Je umístěna na nějakém webovém serveru většinou spolu s webovou aplikací někde na internetu. Podobně jako databáze Varia je databáze obchodu chráněna proti přístupu k datům pomocí autorizace pomocí rozhraní pro výměnu dat.

2.2.6 Výměna dat mezi databází (jednosměrná, obousměrná)

Jednosměrná výměna je dat jednodušší, bezpečnější než obousměrná komunikace. Přístup do firemní sítě z internetu a tím pádem i k databázi zakázán. V tom případě aplikace pro výměnu dat musí být umístěna uvnitř firemní sítě. Takový typ způsobu výměny dat má jisté nevýhody, mezi které patří zbytečné časté dotazy v nastavených intervalech, kdy se dotazují například na nové objednávky obchodu, když nevznikly žádné nové za dobu od předchozího dotazu. A další nevýhodou je nemožnost zjištění stavu skladu a objednávky ze strany internetové obchodu.

Obousměrný typ výměny dat je sice složitější, ale zato je možno implementovat plně automatizovanou výměnu dat s okamžitou reakcí na nově vzniklé položky objednávek, zákazníků atd. a bez zbytečných dotazů na databázi.

2.2.7 Program pro výměnu dat

Pro naprogramování aplikace lze zvolit různé technologie a řešení. Nejčastěji používaným řešením, o jehož popularitě jsem se dozvěděl pomocí různých publikací na internetu je výměna dat ve formátu XML dokumentu pomocí HTTP protokolu, avšak zaleží čistě jen na programátorovi, jaký způsob zvolí. Takový způsob lze pak naprogramovat v různých jazycích na různých softwarových platformách. Akorát program na straně Varia musí umět pracovat s databází MS SQL nebo Access. Na straně obchodu zaleží na platformě, kterou sever využívá. Mezi ně patří ASP, PHP, .NET, JAVA. Také mě zaujalo možnost řešení komunikace využívající pouze Transact SQL s použitím komunikace pomocí propojení MS SQL serverů technologií Linked server.

Principiálně výměna dat většinou funguje tak, že aplikace zajišťující přenos dat je umístěna na nějakém serveru ve firemní síti a bude mít přístup k databázi Altus Varia. Altus Vario v tom případě nemusí být na serveru spolu s jeho databází nainstalované. Program bude odesílat data na server internetového obchodu, kde budou dále zpracovávány do databáze obchodu. Pokud bude přenos fungovat na řešení obousměrné komunikace, měla by aplikace naslouchat na předem stanoveném portu, aby mohly být zpracovány požadavky zvenčí, konkrétně požadavky obchodu pro uložení nových objednávek do Varia nebo zákazníků atd. V případě obousměrné komunikace musí být server zabezpečen, uložen v DMZ zóně firemní sítě. Chráněn firewallem před hrozbami z internetu a být nastaven tak aby mohly přistupovat do sítě pouze určené aplikace komunikující přes konkrétní protokoly jako je například http a plno jiných bezpečnostních ošetření, které by na správném serveru neměly chybět. Pro obousměrnou komunikaci a příjem volání obchodu se hodí komunikace pomocí knihovny Winsock, nebo místo programu se lze vytvořit ASP .NET aplikaci běžící na IIS webovém serveru, také se mohou volat vzdálené metody přes .NETremoting.

Zapisovat data do databází lze buď přímým zápisem do tabulek SQL serveru nebo vyžitím objektových knihovny VarioLib.DLL(COM). Při zápisu do tabulek vždy nutné dodržet předepsané hodnoty (datové typy).

Práce s daty nám může zjednodušit knihovny Varia, ze kterých jsou k mání všechny jeho vlastnosti jako například nastavení výchozích hodnot nových záznamů prostřednictvím šablon, automatické řízení číselné řady dokladů, udržování integrity dat atd. Nevýhodou je rozsáhlost těchto objektových knihoven, což je nutné mít důkladně nastudované pro reálné využití a také, že vyžaduje pouze platformu Windows. Z tohoto hlediska pro přesně definovanou množinu dat bude efektivnější zvolit přímé čtení a zapisování dat do databází.

Za zmínku také stojí, že Altus Vario vyvíjí a nabízí instalační balíček s webovou službou nazvanou Altus Vario Integration Services. Tato služba slouží nejen ke komunikaci aplikací dalších výrobců, ale také určen pro výměnu dat pro internetové obchody, což je největší prioritou této služby.

2.2.8 Obchodní procesy

Data vznikají jak na straně internetového obchodu, tak na straně Varia a proto předpokládáme následující rozdělení rolí:

- V Altus Variu probíhá kompletní manipulace dat s položky katalogů a s tím i související pohyby na skladech, úprava ceníků, knih, vystavují se v něm faktury
- Nové objednávky vznikají v obchodě
- A zákazníci mohou vznikat v adresáři Varia, tak i v obchodě

Důležitým předpokladem je, kde budou vznikat nové záznamy pouze zákazníku, jelikož vznik záznamů z bodu 1 a 2 jsou dostatečně jasné. Není třeba řešit riziko vzniku duplicitních záznamů, jelikož záznamy z 1. a 2. bodu vznikají každý na jednom místě. Ale záznamy zákazníku mohou vznikat na straně obchodu i Varia zároveň, proto je potřeba navrhnout aplikační logiku tak, aby nedocházelo k těmto duplicitám. Dejme tomu, že nově vzniklý zákazník registrovaný přes obchod se prvně uloží do databáze Varia a za předpokladu úspěšného uložení se záznam zkopíruje i do databáze obchodu.

Co se bude dít na straně Altus Varia:

- Práce s položky Katalogů
- Ceníků
- Manipulace se stromovou strukturou kategorií produktů, spárováním jejich identifikací (id řádků) a identifikací produktů se vytvářejí položky kategorií
- Vyřizování zakázek obchodu
- Vystavování faktur zakázek obchodu a vydávání zboží

Tyto záznamy budou vznikat pouze a možnost je měnit pouze na straně Varia odkud je bude třeba odesílat do obchodu po každé změně, v určeném intervalu, nebo také v kombinaci obou případů. To už zaleží na programátorovi.

Co se bude dít na straně obchodu:

- Ideálně by se měl zjišťovat aktuální stav skladu ve Variu produktů, pokud to umožňuje obchod za předpokladu obousměrné komunikace
- Vystavování objednávek obchodu (pro Vario se používá termín zakázky)
- Registrace nových zákazníků
- Zjišťování stavu objednávek

Při použití obousměrné komunikace má funkcionalita obchodu možnost zjišťovat stav skladu, jinak se musí pravidelně zasílat aplikaci z Varia do obchodu v případě jednosměrné komunikace. Nové objednávky se budou zasílat do Varia a ukládat do knihy agendy Zakázky (nejlépe s názvem knihy třeba „Zakázky eshop“ pro odlišení ostatních zakázek).

2.2.9 Hotová řešení

Hlavně kvůli velké referenci Altus Varia existují firmy, které se zabývají propojením obchodu a Varia a mimo jiné vytvářením internetových obchodu na míru. Většinou se pak zaměřují nejen na jeden ERP systém, ale jejich nabídka pak obsahuje k výběru hned několik takových systémů a kromě toho i výběr z několika variant balíčků podle cenových kategorií, které obsahují, či jsou ochuzeny o různé služby.

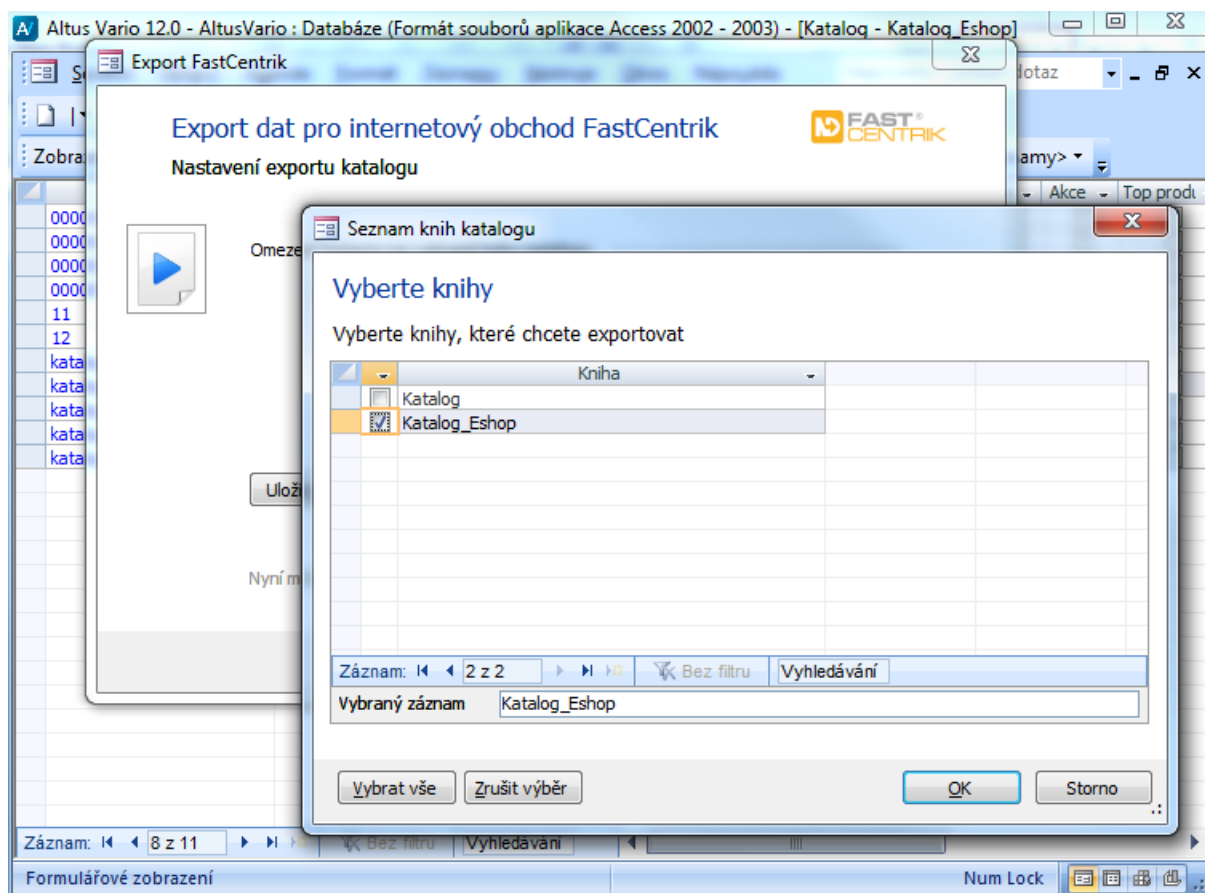
Mezi tyto firmy patří například:

- NetDirect ShopCentrik - internetové obchody jsou vytvářeny zákazníkům podle jejich potřeb na míru a jsou postaveny na systému ShopCentrik. Nespornou výhodou je, že ShopCentrik využívá obousměrnou komunikaci a je plně automatizován
- NetDirect FastCentrik - jak už lze z názvu odvodit je určený pro zákazníky, kteří chtějí začít obchodovat co nejrychleji (do 24 hodin). Nabízí různé jazykové provedení, bezplatné aktualizace rozšiřující funkcionalitu, školení i telefonická podpora a další.
- ShopSys - je profesionální WEB 2.0 aplikace pro obchodování po internetu. Tyto internetové obchody se vyznačují intuitivním nakupováním a snadnou správou v administračním rozhraní. Lze na nich vystavět B2B portály či patrová internetová vícejazyčná nákupní centra s možností platit v různých měnách stejně jako jednoduché malé internetové obchody se zaměřením na koncový prodej jednoho sortimentu.

A plno dalších existujících řešení pro internetové obchody jako je Jedničky – internetové obchody na míru, Nový Web s.r.o., web2way, CZI.

2.2.10 Založení knih

Pro lepší přehlednost, filtraci dat přímo pro internetový obchod je nutné si první založit knihy potřebných agend. Název by měl jednoznačně rozlišit záznamy internetového obchodu od ostatních, například název knihy pro agendu katalog by se měl být „Katalog eshop“. Tak samo se jedná o agendy adresář, sklad, ceníky a další. Názvy knih se využívají při exportu dat do XML souborů nebo se dotazovat pomocí SQL dotazů, kde lze výběrem z knih Katalogu manipulovat s produkty těchto vybraných knih a se souvisejícími položky skladů, ceníků.



Obrázek 4.2: Vario – export pro FastCentrik, výběr exportovaných knih katalogu

2.2.11 Pořadí importovaných souboru, dat

Při importování dat z XML souboru a tím je myšleno celkově vkládání dat jako takové je nutné zachovat pořadí importovaných agend a tomu je jak při importu na straně varia tak in straně obchodu. Tabulky databázi obsahují cizí klíče jiných tabulek (mají mezi sebou vazby). To znamená, že nemůžu importovat dříve ceník, který bude obsahovat ceny nových produktů, aniž bych před tím neimportoval nové produkty související s těmito cenami. Pokud se tak stane, program to bude brát jako chybu hlášenou zprávou typu „Produkt s takovým klíčem se nenachází v Katalogu“. To samé platí i pro manipulaci s existujícími položky tabulek, které budeme chtít mazat, upravovat. Lepší volbou je označit položku příznakem Smazáno a dále tuhle položku nezobrazovat. Nejde smazat položku katalogu, na kterou se odkazuje položka dokladu objednávky, skladu, ceníku atd.

2.3 Nástin možných řešení

2.3.1 Výměna dat přes XML rozhraní

Hodně rozšířeným řešením pro přenos dat mezi databázemi se používá přenos dat na základě dokumentu v XML formátu. Data na straně z jednoho systému se exportují do XML souborů (například katalog bude ve tvaru katalog.xml), následně se odešlou do datového úložiště druhého serveru a tam se data z XML souboru zpracují a importují do odpovídající databáze.

Stručně popsany příklad: Export dat do XML z databáze Varia → přenos xml souboru ze serveru Varia přes FTP klienta do předem určené složky na server internetového obchodu → import XML souboru pomocí webového rozhraní do databáze internetového obchodu. Obdobně se provede export a import opačně z internetového obchodu do Varia.

XML dokument

XML(Extensible Markup Language) je značkovací jazyk, v tomto případě použit pro výměnu dat mezi aplikacemi. XML je jazyk navrhnutý tak, aby mohl uchovávat strukturovaná data. Vezmeme si například nějakou HTML stránku nějaké firmy, která zde prezentuje své výrobky či služby. Pro člověka jsou všechna data jednoduše čitelná. Pokud například uvidíme adresu firmy, poznáme, že se jedná o její adresu.

Pomocí XML je možno strukturovat data tak, aby byla použitelná při databázovém zpracování. Jazyk XML má samozřejmě další možnosti. Ať už se jedná o výměnu dat mezi pobočkami firem (business-to-business applications), webové stránky, elektronické publikování či úložiště dat pro programy.

Umožňuje snadné vytváření konkrétních značkovacích jazyků (tzv. aplikací) pro různé účely a různé typy dat. Zpracování XML je podporováno řadou nástrojů a programovacích jazyků. Jazyk je určen především pro výměnu dat mezi aplikacemi a pro publikování dokumentů. Popisuje strukturu z hlediska věcného obsahu jednotlivých částí, nezabývá se vzhledem.

XML dokument je text, vždy Unicode, v Česku obvykle kódovaný jako UTF-8, ale jsou přípustná i jiná kódování.

Syntaxe XML - příklad:

```
<?xml version="1.0" encoding="UTF-8" ?>
<katalog>
  <nazevPolozky>Monitor LG</nazevPolozky>
  <text>Velice pěkný monitor, nízká spotřeba, vysoké spektrum
    barev atd.</text>
  <vyrobce>LG</vyrobce>
</katalog>
```

FastCentrik využívá k validaci XML dokumentů XSD schémat, což je schéma využívající XML formát pro popis struktury XML dokumentu.

XML schéma:

- definuje místa v dokumentu, na kterých se mohou vyskytovat různé elementy
- definuje atributy
- definuje, které elementy jsou potomky jiných elementů
- definuje pořadí elementů
- definuje počty elementů
- definuje, zda element může být prázdný, nebo zda musí obsahovat text
- definuje datové typy elementů a jejich atributů
- definuje standardní hodnoty elementů a atributů

Použití:

- popisuje přípustný obsah dokumentu
- umožňuje kontrolovat správnost dat
- umožňuje pracovat s daty uloženými v databázi
- umožňuje konvertovat data mezi různými datovými typy

Příklad demonstřuji na příkladě z článku o XML dokumentu

XSD dokument bude vypadat následovně:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="katalog">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="nazevPolozky"
          type="xs:string"/></xs:element>
        <xs:element name="text" type="xs:string"/></xs:element>
        <xs:element name="vyrobce" type="xs:string"/></xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Připojení XSD k XML dokumentu:

```
<?xml version="1.0" encoding="UTF-8"?>
<katalog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="katalog.xsd">
  <nazevPolozky>Monitor LG</nazevPolozky>
  <text>Velice pěkný monitor, nízká spotřeba, vysoké spektrum
    barev atd.</text>
  <vyrobce>LG</vyrobce>
</katalog >
```

Import a export XML souboru pro FastCentrik

Vario obsahuje jednoduchý nástroj pro výměnu dat s internetovým obchodem FastCentrik. Toto propojení výrazným způsobem usnadňuje správu internetového obchodu, protože lze spravovat data pouze ve Variu a e-shop aktualizovat pomocí exportů. Zároveň je možné opačným směrem z obchodu do Varia přenášet internetové objednávky a kontakty.

I když je tento export do XML souboru určený výhradně pro systém Fastcentrik, jeho export a import se dá přizpůsobit vlastním potřebám pro implantaci vlastního webového rozhraní pro výměnu dat.

FastCentrik je B2C řešení určené koncovým spotřebitelům. Disponuje rozsáhlou funkcí a jednoduše se obsluhuje. Vzájemná výměna dat mezi e-obchodem FastCentrik a ekonomickým systémem Altus Vario je realizována pomocí exportu a importu, které Altus Vario, ať už v plnohodnotné nebo zjednodušené verzi, obsahuje v základu zdarma.

FastCentrik je e-shop spravovaný uživatelským způsobem. Abyste data do obchodu nemuseli zadávat ručně, FastCentrik podporuje import dat. Importovat lze:

- produkty, související produkty, alternativní produkty
- kategorie produktů
- ceny, ceníky
- parametry produktů
- elektroodpad
- informaci o skladové dostupnosti
- obrázky
- měrné jednotky
- emaily pro zasílání novinek
- slevy výrobce
- feed pro vyhledávač zboží.cz

Export maximálním způsobem využívá data zadaná ve Variu a minimalizuje nutnost složitě nastavování. Výstupem exportu je několik datových souborů ve formátu XML, které je možné v administraci obchodu importovat a naplnit tak obchod.

Export

Po spuštění exportu z agendy Katalog se provede načtení dat, a pokud data obsahují nějaké záznamy, proběhne validace a data se uloží do XML souboru s přesně definovaným formátem. Pokud by XML soubor neobsahoval žádná data, vůbec se nevytvoří. Validace XML dokumentu probíhá na základě XSD šablon poskytnutých od provozovatele internetového obchodu. Pokud neproběhne validace některého ze souborů v pořádku, export na závěr otevře seznam validačních chyb.

Import

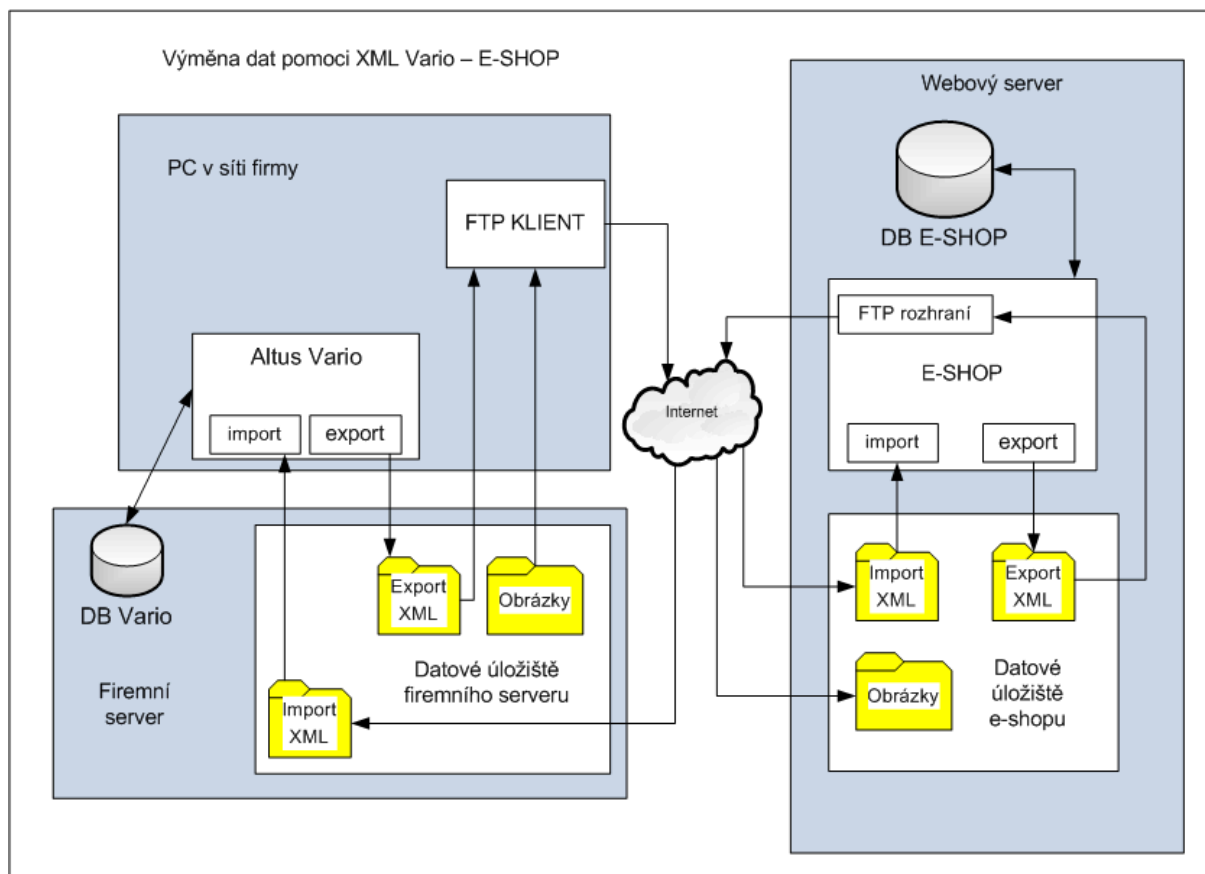
Import spustíte z agendy Adresář nebo Zakázky. Otevře se dialog s nastavením, které lze uložit jako samostatný import.

Validace a bezpečnost

V současné době je tato komunikace upřednostňována před importy a exporty tabulek i přímým přístupem do databáze. Důvodem je vyšší bezpečnost operací, neboť při XML komunikaci dochází ke kontrole a validaci dat tak, jako by šlo o operace vykonávané přímo v prostředí programu. Např. při importu dokladů ve formátu XML tak dochází ke kontrole vstupních údajů, dopočítání chybějících informací a v případě chybovosti importovaných dat také k upozornění na nutnost doplnění nebo přepracování XML souboru bez jeho importování. Bezpečnost této formy komunikace je srovnatelná s bezpečností a možnostmi ručního zápisu dat v prostředí programu Altus Vario.

Nevýhody:

- Při exportu využitím FastCentriku si nelze navolit svoje vlastní položky, které bych chtěl importovat do obchodu
- Musím vytvářet zvlášť export pro adresář a pro jiné agendy



Obrázek 4.3: Výměna dat exportem a importem do XML dokumentu, pro FastCentrik

Výměna dat – postup:

1. V Altus Variu jsem si otevřel agendu katalog.
2. Výběrem Soubor > Export dat.. > Export FastCentrik jsem otevřel průvodce exportem dat XML dokumentu pro FastCentrik.
3. V průvodci jsem zvolil složku pro exportované dokumenty, ze které se pak FTP klientem odesílaly tyto soubory do datového úložiště e-shopu složky určené pro import.
4. Nakonec se přes webové rozhraní načetly data z těchto XML souboru a zpracovaly do databáze Varia.

Tak se provádí obdobně export dat z internetového obchodu do Varia. Pokud bych chtěl vytvářet nové zákazníky v Altus Variu a exportovat je do databáze obchodu, musel bych zvlášť provést export pro zákazníky a uložit XML do složky určené pro export FTP klientem do složky v obchodě.

Import objednávek a zákazníků z internetového obchodu se také neprovádí moc automatizovaně.

1. Je třeba provést export v rozhraní obchodu.
2. Soubory se pošlou do složky určené k importu do databáze Varia.
3. Musí se obě agendy importovat ručně a zvlášť, hlavně ve správném pořadí. První zákazník kdyby byl náhodou nově registrovaný s nově vytvořenou objednávkou.

Ale tohle řešení se mi nijak nezamlouvá. Moc klikání, přeposílání dokumentu na jiná místa, hodně zdoluhavé. Takové interakce vyžaduje i zkušenější uživatele, což by mohl být další problém, hlavně při ovládání FTP klienta.

Po delším zamýšlení a nabrání nových zkušeností přečtením informací například webových stránek, které se zabývají užitím technologií pomocné pro výměnu dat mezi ERP systémy a internetovými obchody, mě nakonec napadla schůdnější varianta. Proč si nenapsat pomocnou aplikaci, kterou by uživatel ovládal spolu s Variem.

Aplikace by měla tyto vlastnosti:

1. Obsahovala by tlačítko Obnovit zboží, Obnovit zákazníky, tím by se kompletně bez dalších interakcí uživatele provedla výměna dat z databáze Varia do internetového obchodu.
2. Obsahovala by Timer, to znamená, že výměna dat by se prováděla automaticky v nastaveném intervalu a pouze pokud by aplikace byla spuštěná

Jak by to probíhala výměna:

1. Poklikáním na kterékoliv Obnovit tlačítko aplikace vygeneruje XML soubory. Příkladem může být naplnění datasetu s potřebnými tabulkami agend jako je Katalog, Ceníky, Sklady aj. Dataset pak lze jednoduše přenést do XML dokumentu.
2. Druhým krokem by aplikace pomocí nastaveného FTP rozhraní odeslala dokument do zvolené složky v datovém úložišti internetového obchodu
3. Aby se mohl XML dokument na straně obchodu zpracovat a provést se výměna dat, musela by se vyvolat nějaká událost, která by spustila metodu na rozhraní obchodu a obnovila a provedla výměnu dat. Toho lze dosáhnout pomocí technologie .NetRemonig, neboli vzdálené volání metod. Takže abych to shrnul, po přenesení XML dokumentu se zavolá metoda, na straně obchodu provede výměnu dat a ještě vrátí zprávu o úspěšnosti akce.

Obdobně by výměna dat probíhala ze strany obchodu do Varia.

2.3.2 Altus Vario Integration Services

Altus Vario obsahuje Altus Vario Integration Services(AVIS) což je webová služba, která komunikuje přes komunikační webové rozhraní. Z toho lze odvodit, že AVIS slouží pro komunikaci, výměnu dat s jinými systémy a funguje na bázi SOAP.

Webové služby

Webová služba umožňuje webové stránce rozšířit její funkcionalitu a dynamičnost. Dnes je možno přes internet provádět dotazy nad databázemi, objednat si letenku, zjistit stav naší zakázky apod.

Z pohledu klienta či zákazníka je webová služba nějaký prostor na webu, ve kterém se vykonávají operace volané zvenčí. Pod každou takovou operaci si v našem případě můžeme představit jako čtení nebo zápis dat, například přidání nového zákazníka nebo čtení katalogových položek. Operace můžou, ale také nemusí obsahovat nějaké vstupy, ať už jednoduché, např. Id řádku katalogu, nebo také seznam položek objednávek určená k zápisu. Podobně má také pak operace i svoje výstupy.

Webové služby se také dají využít jako základní stavební bloky pro vytváření aplikací distribuovaným přístupem dat. Aplikaci jsou pak tvořeny k využívání více různých webových služeb a různých zdrojů bez ohledu na to, jak služby byly implementovány a z jakého místa jsou provozovány. Mohou být volány prostřednictvím POST nebo GET metod protokolu HTTP za pomoci protokolu SOAP. Dokážou komunikovat s programy napsány v různých jazycích na různých platformách standardním způsobem, což je bezesporu velká výhoda.

SOAP, WSDL, UDDI

SOAP (Simple Object Access Protocol) je protokol sloužící pro vzdálené volání procedur anglicky řečeno Remote procedure call (RPC). SOAP je založený na výměně XML zpráv přes internetovou síť pomocí HTTP protokolu pro výměnu dat hypertextových dokumentu ve formátu HTML. Jednoduše řečeno pomocí HTTP protokolu probíhá veškerá komunikace webových serverů a prohlížeči webových stránek. Výsledkem nám je vždy právě webová stránka jak ji známe.

Aby bylo snadnější psaní uživatelských aplikací jsou webové služby psané ve strojově zpracovatelném formátu, konkrétně WSDL(Web Services Description Language - jazyk popisu webových služeb), který popisuje rozhraní webové služby, tedy sadu SOAP zpráv a způsob jakým se vyměňují. Specifikuje obsah zprávy s typy argumentu a také její výstup s návratovou hodnotou v jednoznačném zápisu. WSDL se dá číst a editovat, ale nejčastěji je automaticky generován a využíván softwarem. WSDL spolu se SOAP jsou provedeny v syntaxi XML jazyka.

UDDI specifikace zajišťuje registrování webových služeb, pro jejich snadnější nalezení na internetu. Také ho lze přirovnat ke zlatým stránkám webových služeb, na kterých lze hledat firmy nabízející požadované služby, přečíst si informace o nabízejících službách a následně po vybrání některé z nich kontaktovat firmu pro další informace. Webovou službu lze samozřejmě nabízet bez registrace v UDDI, pokud ji chce využít třeba jen pro své vlastní účely.

Avis je tedy vhodný pro navrhnutí sdílení dat nejen pro systémy internetových obchodů, ale i například pro komunikaci s on-line čtečkami čárových kódů a jiných informačních systémů.

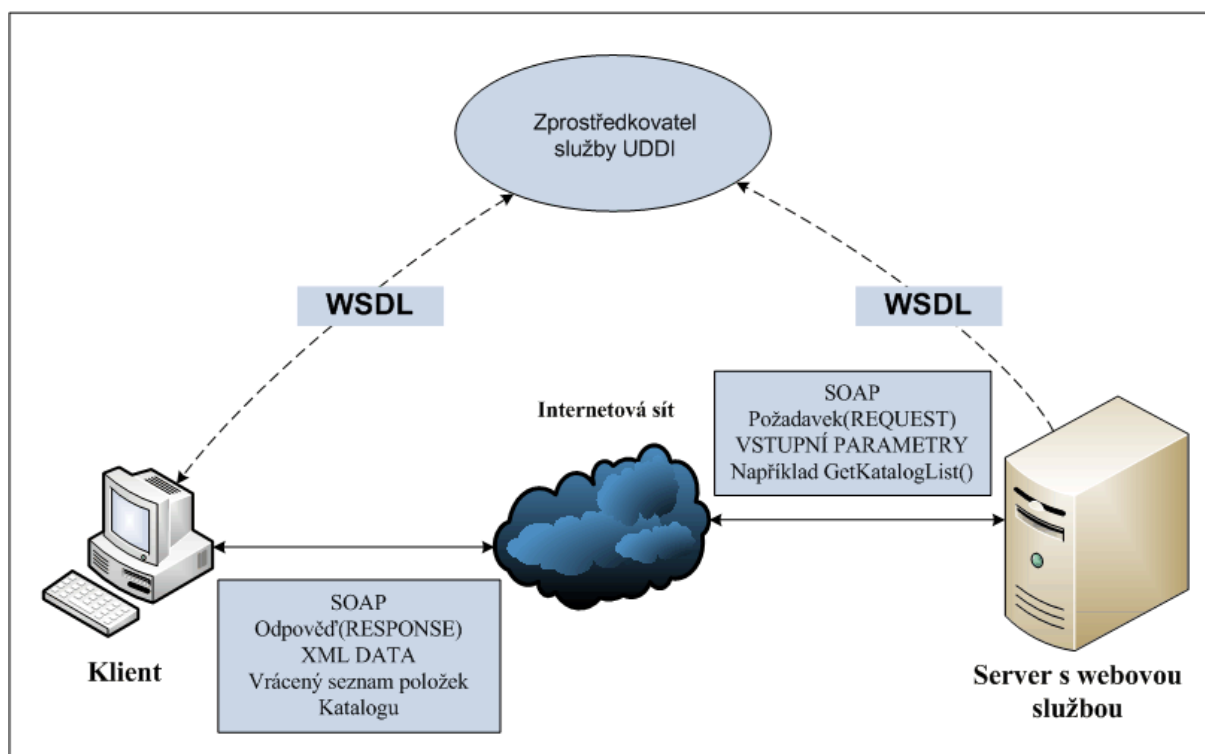
K provozování AVIS je potřeba Windows server a připojení k internetu. Tato webová služba po spuštění na serveru pak komunikuje s Altus Variem skrz velké množství metod, které mohou využít veškeré druhy externích aplikací jednak pro získání potřebných informací, dat z katalogu, skladu atd., ale také pro jejich ukládání třeba nových objednávek, kontaktu.

Příklady získávání a zadávání dat mezi Altus Variem a internetovým obchodem:

- Zadávání dokladů do Varia (zakázky, faktury, skladové doklady),
- Rezervace položek na skladě,
- Vložit / načíst kontakt do adresáře,
- Načíst katalogové položky,
- Načíst aktuální stav položek na konkrétním skladu,
- Načítání aktuálních ceníků,
- Načtení vytvořeného stromu produktů (kategorií).

Mezi výhody této webové služby AVIS patří to, že externí aplikace či systém nemusí přímo přistupovat k databázi, ale všechno za ní udělá AVIS. Dokonce zajistí, aby změny provedené v databázi Varia byly okamžitě viditelné a zadávána data byla korektně a správně ošetřena před uložením do databáze. Mezi další výhody určitě patří vlastní rozšíření dosavadních, nebo přidání nových metod. Pokud bych zrovna nenašel, co potřebuji.

Abych mohl webovou službu AVIS užívat, tak musím jako zdroj dat při instalaci zvolit MS SQL, což si myslím není žádný problém, jelikož SQL Server 2008 R2 Express je zdarma a pro provoz a ostrý provoz systému dostačující.



Obrázek 4 .4: Znáznornění použití webových služeb

Jak vlastně volání takové webové služby probíhá:

- Po zavolání vzdálené procedury se parametry a identifikátory procedury zabalí do vhodné formy pro přenos přes internetovou síť a balíček se odešle jako požadavek (anglicky REQUEST)
- Na vzdáleném místě se balíček rozbalí a zjistí se, o jakou proceduru jde, která se následně vykoná
- Následně se získaná data zabalí, například seznam katalogových položek a odešlou se zpět na místo jako odpověď (anglicky RESPONSE), odkud byla procedura volána a data mohou dále dle libosti použita
- Pokud procedura není určena ke vrácení dat, ale například k uložení objednávky do databáze Altus Varia, odešle se zpět aspoň zpráva o úspěšnosti vykonání procedury

Na závěr webových služeb jsem chtěl dodat, že tohle řešení je mým favoritem. Během testování, jsem narazil na některé komplikace a to že nefungují ještě na sto procent. Tohle řešení je ještě v plenkách a také mě trochu odradila složitá instalace.

2.3.3 Přímé propojení databází

Přímým propojením databází je myšlena výměna data pomocí Transact-SQL. Oba zdroje dat jak Varia, tak internetového obchodu musí být MSSQL databáze a každá z nich bude na svém serveru. Databáze pak lze propojit vzdáleně pomocí technologie Linked serveru s následnou možností dotazování, volání procedur atd. z databáze serveru A do databáze serveru B, takže se vše nakonec chová podobně jako na jediném MSSQL serveru.

SQL (Structured Query Language) je non-procedurální jazyk pro dotazování a změny dat v databázi. SQL umožňuje definovat, vybírat, upravovat data, a rovněž umožňují řízení přístupu k nim. SQL příkazy mohou být vloženy do programů psaných v jiných jazycích, jako C / C ++, Java, PHP, Python. SQL zahrnoval několik složek, z nichž nejdůležitější jsou: DDL - data jazyk pro popis dat DML jazyk pro manipulaci.

Transact-SQL (T-SQL) je rozšíření SQL od společností Microsoft a Sybase. Microsoft tento jazyk používá v produktu Microsoft SQL Server. T-SQL rozšiřující standardní SQL zahrnuje procedurální programování, lokální proměnné, různé podpůrné funkce pro zpracování řetězců a datumů, nové matematické funkce.

```
CREATE PROCEDURE NazevProcedury (@parametry...)
as
BEGIN
    -- volání jiné procedury
    -- tělo procedury (select, insert, delete atd...)
END;
GO
```

Už jsem se zmínil že pomocí Linked serveru lze propojit dva vzdálené MS SQL servery, na kterých jsou databáze a pak se mohou provádět manipulace s daty, volání procedur na vzdáleném serveru ze serveru, na kterém je vzdálené spojení vytvořeno.

Linked server s technologií OLE DB

Jenže Linked servery neslouží jen ke vzdálenému propojení serveru, ale také k připojení jiných non-databáze datových zdrojů, mezi které patří například ODBC, Access, Oracle databáze pomocí technologie zvané OLE DB.

Technologie OLE DB je vysoce výkonná databázová technologie, založená na modelu COM, vyrobená společností Microsoft. To, co odlišuje technologii OLE DB od jiných databázových technologií společnosti Microsoft, je její poskytování univerzálního přístupu k datům.

Univerzální přístup k datům

Univerzální přístup k datům umožňuje společný přístup k datům, bez ohledu na formu, ve které je uložen. Může se jednat o data uložené přímo v souborových systémech (FAT, NTFS), indexovaných sekvenčních souborech, v malých osobních databázích (například aplikace MS Access), souborech typu Excel nebo v aplikacích projektového plánování jako je MS Project.

Přístup k těmto datům pomocí různých přidružených aplikací představuje hlavní problémové místo v pracovním postupu nebo přinejmenším v zabezpečení. Většina společností se ocitne v této situaci a řeší potíže sloučením informací do databázového systému (DBMS). Avšak takový krok je nákladný, časově náročný a v mnoha případech nepraktický.

Alternativou je vyvinutí řešení univerzálního přístupu k datům. Technologie OLE DB a rozhraní ADO, poskytují možnosti univerzálního přístupu k datům. Z těchto dvou je technologie OLE DB výkonnější a je doporučena pro použití s aplikacemi Visual C++.

Univerzální přístup k datům zahrnuje dvě možnosti: první je distribuovaný dotaz nebo jednotný přístup k více (distribuovaným) zdrojům dat a druhou je schopnost zpřístupnit non-DBMS zdroje dat databázovým aplikacím.

Distribuovaný dotaz

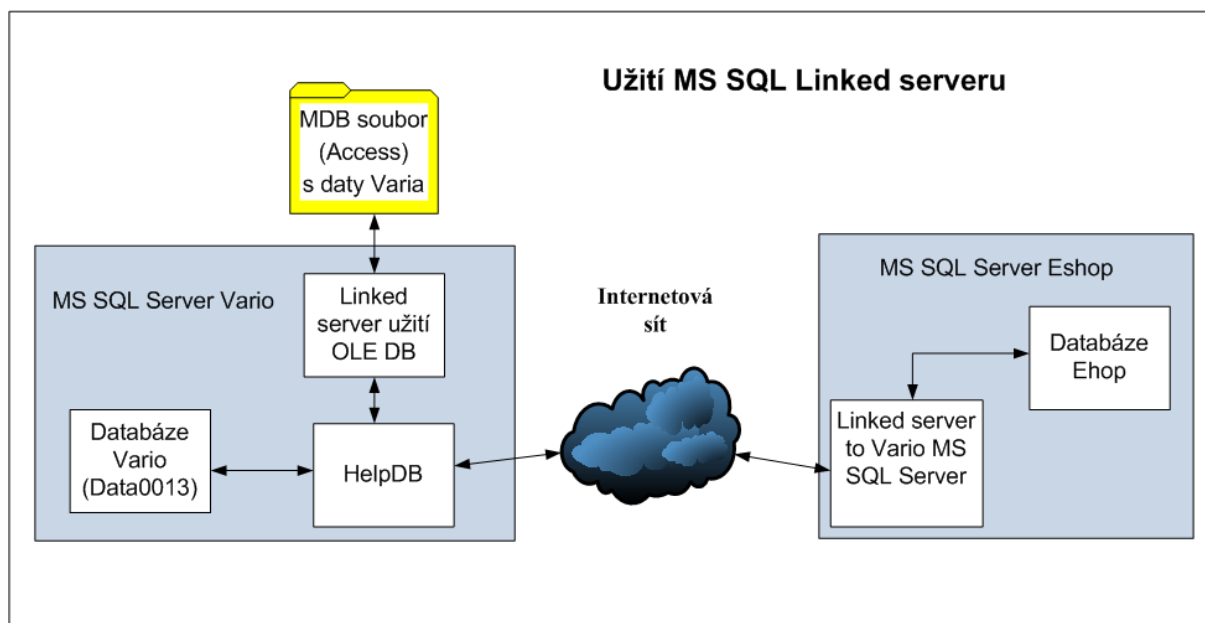
Schopnost získat přístup k datům jednotně z více (které jsou distribuovány) zdrojů dat. Zdroje dat mohou být stejného typu (například dvě samostatné databáze aplikace Access) nebo různých typů (například databáze serveru SQL Server a databáze aplikace Access). Jednotně znamená, že lze srozumitelně spouštět stejný dotaz na všechny zdroje dat.

Non-DBMS přístup

Schopnost zpřístupnit non-DBMS zdroje dat databázovým aplikacím. Příklady DBMS zdrojů dat jsou DB2, Oracle, SQL Server, Access, Sybase Adaptive Server Enterprise, FileMaker, Firebird. Příklady non-DBMS zdrojů dat zahrnují informaci v systémech souborů, emailech, tabulkových procesorech a nástrojích projektového řízení.

Technologie OLE DB umožňuje vyvíjet aplikace, které přistupují k různým zdrojům dat, ať už jsou DBMS nebo ne. Technologie OLE DB umožňuje univerzální přístup pomocí rozhraní modelu COM, který podporuje příslušné DBMS funkce pro daný zdroj dat. Model COM snižuje nežádoucí zdvojení služeb a maximalizuje vzájemnou funkční spolupráci nejen mezi zdroji dat, ale také mezi jinými aplikacemi.

Jelikož Altus Vario nabízí při instalaci výběr zdroje dat mezi MS SQL databází a Access, je nakonec na programátorovi nebo zadavateli jaký zdroj dat zvolí a výsledek bude zcela totožný, protože v každém případě je potřeba mít jak na serveru Varia, tak na serveru internetového obchodu MS SQL server nainstalovaný. A proč tomu tak musí být, naznačuje schéma na následném obrázku.



Obrázek 4 .5: Užití MS SQL Linked servru

Pár slov ke schématu

Na obrázku je vidět dva zdroje dat Altus Varia a to je jen pro příklad, jak by to fungovalo kdyby, kdybych si zvolil jeden z nich. Samozřejmě se použije jen jeden a to buď databáze a nebo mdb soubor. Podle mě je lepší variantou mít nainstalovaný zdroj dat přímo na MS SQL serveru spolu s HelpDB, kterou popíši následovně.

K čemu je vlastně tahle HelpDB databáze?

Tahle databáze je na chlup stejná jako databáze internetového obchodu.

Důvodů existuje hned několik:

- Jedná se o důvěru zákazníka. Altus Vario obsahuje mnohem více informací, které jsou zbytečné pro provozování internetového obchodu a také citlivé pro majitele dat. Programátor tak bude pracovat jen s daty, které potřebuje.
- Do téhle databáze se ukládají nová, pozměněná nebo smazána data a tyhle zpracované data dále internetový obchod stahuje k sobě do databáze.
- Do databáze Varia (a už vůbec nemožné v souboru mdb) není vhodné vytvářet procedurální logiku (procedury, trigery atd.) jelikož by to nebylo v souladu s prvním bodem a na druhou stranu při aktualizaci Varia a jeho databáze, bych mohl o tuhle rozšířenou funkcionalitu databáze přijít.

Výhody

Za jisté výhody považují u téhle metody přímého propojení databázi jednoduchá implementace, pokud znám strukturu databáze Varia. Mohu tak vyžívat všechna data uložená v databázi a nejsem omezen funkcionalitou programu Varia k exportu a importu dat.

Nevýhody

Mezi nevýhody patří změna struktury databáze s další příchozí aktualizací Varia zasahující právě do částí tabulek využívající internetový obchodu. Autoři si ve většině případu tyto změny nechávají pro sebe.

2.4 Výběr vhodného řešení

Náplní této kapitoly byl nástin a návrh možných řešení propojení Altus Varia 12 s internetovým obchodem. Po důkladné analýze, která byla v mých silách, a prostudování dokumentace Varia jsem celkem možná řešení. Tato řešení jsem částečně testoval, konzultoval s vedoucím bakalářské práce a nakonec dospěl k následujícím závěrům. Hlavně bych chtěl podotknout, že největší důraz byl kladen na co největší automatizaci, aby uživatel nemusel provádět nejlépe žádnou interakci při výměně dat.

S takovým požadavkem XML rozhraní neobstálo a to z těchto důvodů:

- Import a export dat byl děláný výhradně pro systém FastCentrik a to takovým způsobem, že v prvním kroku bylo data potřeba manuálně ve Variu exportovat do XML a tyto XML soubory pak nějakým způsobem dostat na server internetového obchodu, kde by se musely přes nějaké rozhraní zpracovat. Struktura dat by možná odpovídala, ale to jsem dále nezkoušel.
- Jako další možnost by byla aplikace (třeba i nějaká služba) vytvořena programátorem běžící na počítači spolu s Variem, která by dohlížela na export dat.

Webové služby mají v dnešní době velkou budoucnost. O tom vědí i vývojáři Altus Varia. I když je instalace náročná, určitě to pak stojí za to. Není ohrožena integrita dat a přístup k datům je zajištěn odkudkoliv. Leč i od této varianty jsem musel upustit, jelikož jsou teprve ve vývoji a nešlapou tak jak by měly. Mají ještě své mouchy.

Nakonec jsem se rozhodl pro výměnu dat přímým přístupem k databázi. I když nemám jistotu ve struktuře dat s další novou verzí Varia a dokumentace databáze je také nedostatečná. Na druhou stranu mohu využít všechna data uložená v tabulkách, nejsem omezen funkcionalitou programu Vario a po důkladné analýze tabulek dělá z řešení přímého přístupu k databázi silný nástroj.

3 Definice datového rozhraní pro přenos dat z ERP.

Jak už jsem nakouzl v minulé kapitole, nakonec jsem zvolil pro propojení Altus Varia s internetovým obchodem pomocí přímého přístupu k databázi. Takže nebude v žádném případě problém, pokud zvolím jako datový zdroj Varia MS SQL databázi nebo Access. V obou případech stejně musí být ve firemní síti MS SQL server nainstalován, abych mohl načítat data pomocí technologie OLE DB z Access mdb souboru nebo abych mohl nechat vytvořit databázi dat pro ERP systém.

Vše co bude použito v datovém rozhraní:

- Databáze pro uchování dat Varia a internetového obchodu
- Procedury pro manipulaci s daty
- Linked server pro propojení MS SQL serveru nebo pro připojení odlišného datového zdroje než je databáze MS SQL

Celkem se bude pracovat se třemi databázemi. I když instalace Varia jich vytvoří více než jednu, pro potřebné informace, které se budou přenášet, bude to zrovna jedna konkrétní databáze s názvem “Data” + čtyřmístné číslo, které se zadává během instalace.

Další druhá databáze, pomocná, obsahuje už jen konkrétní tabulky, atributy nutné pro přenos dat do internetového obchodu, dále procedury, které v žádném případě nemohou být v databázi. Je to takové pravidlo. Nikdy nezasahovat do struktury cizích databází, mohlo by se to vymstít formou nové verze softwaru a tak i zánikem provedených změn.

Poslední databáze je databáze internetového obchodu. Může být umístěná na serveru spolu s webovým serverem nebo i na nějakém jiném místě v síti. Tato databáze je totožná s pomocnou databází na serveru s Variem.

Datový slovník

Největší pozornost zasloužila databáze internetového obchodu či pomocná databáze pro výměnu dat. Tato databáze vyžadovala důkladný rozbor a analýzu funkcionalit Varia, a během tvorby internetového obchodu jsem prováděl často změny nad strukturou dat. Tabulky Varia obsahovaly mnohem větší množství atributů než tabulky internetového obchodu po redukci nepotřebných atributů.

Tabulka.4.1: Seznam tabulek v pomocné či databázi internetového obchodu

Název tabulky	Popis
BankovníSpojení	Bankovní spojení
Ceník	Ceníky položek katalogu
Doklady	Objednávky
Firmy	Evidence osob a organizací
Katalog	Katalog produktů
Kategorie	Strom kategorií
KategorieProdukty	Produkty svázané s kategorií
PoložkyDokladu	Položky objednávek
Sklad	Sklad
SouvisejícíProdukty	Související a alternativní produkty

Seznam atributů, datových typů, popisu jednotlivých atributů a je přiložen v příloze jako datový slovník.

4 Návrh a realizace

V této kapitole popíší software použité k přenosu, procedury určené k synchronizaci dat a vytváření dat a v poslední řadě mnou uskutečněné návrhy a pravidla pro správný chod a výměnu dat.

4.1 Použitý software

Systém pro realizaci propojení bude postaven na relační databázi Microsoft SQL. Pro účel této práce byla zvolena verze SQL Server 2008 R2 Express, která je zdarma. Tato verze je dostačující nejen pro otestování funkčnosti, ale i pro ostrý provoz, kde je schopna zvládnout i větší provoz. Omezení verze Express je ve velikosti databáze, která je u verze 2008 R2 zvýšena na 10 GB. Využije pouze jeden procesor a 1 GB operační paměti. Dále Neobsahuje službu SQL Server Agent a není tedy možné spouštět pravidelně tzv. Joby, které by mohly spouštět např. procedury.

Propojení databázi jsem testoval na jednom MS SQL serveru, na kterém běží všechny potřebné databáze. Tzn. bez propojení serveru pomocí Linked server.

4.2 Uložené procedury

Uložené procedury zajišťují aktuálnost dat pro e-shop a Altus Vario. Rozumí se tím adresy a zboží určené k prodeji včetně informací o cenách. Také aktualizují adresy a vkládají nové objednávky do databáze Varia.

Tabulka.4.2: Výpis uložených procedur e-shopu

Procedury	Popis
KontrolaDokladu	Kontroluje nepřenesené objednávky do Varia
KontrolaPolozekDokladu	Kontroluje nepřenesené položky objednávky do Varia
KontrolaZakazniku	Kontroluje nepřenesené registrace do Varia
OnlinePohledDoklady	Pohled do Varia pro informace objednávky
OnlinePohledPolozkyDokladu	Pohled do Varia pro informace objednávky
PridatDoklad	Nová objednávka
PridatPolozkuDokladu	Nové položky objdnávky
PridatZakaznika	Nová registrace
UpdateCenikHelpEshopDBToEshop	Obnova dat ceníku
UpdateFirmyHelpEshopDBToEshop	Obnova dat zákazníků
UpdateKatalogHelpEshopDBToEshop	Obnova dat produktů
UpdateKategorieHelpEshopDBToEshop	Obnova dat kategorií
UpdateKategorieProduktyHelpEshopDBToEshop	Obnova dat vazební tabulky
UpdateSkladHelpEshopDBToEshop	Obnova dat skladu
UpdateSouvisejiciProduktyHelpEshopDBToEshop	Obnova dat souvisejících produktů

4.3 Výměna dat, postupy a pravidla

Výměnu dat v řešení přímého propojení databází zajišťují uložené procedury. Funguje to tak, že rozhraní internetového obchodu buď pomocí tlačítka, nebo nastaveného intervalu postupně volá procedury a obnoví data první z Varia do pomocné databáze a pak do databáze internetové obchodu zkopíruje jen změněná data.

Kdežto data měněná nebo vytvářená v obchodě se projeví okamžitě v programu Vario neboť se pracuje jen s konkrétním objektem. Jedná se data zákazníku a objednávek. Například objednávka po vytvoření by měla být totiž v co nejkratším možném termínu viditelná ve Variu.

4.4 Aktualizace dat, knihy agend datum aktualizace

Systém je důležité mít dobře navrhnutý hlavně, aby co jak nejméně okrádal zdroje především SQL přístupu a přenosu dat. Proto jsem se snažil optimalizovat přenos na minimum přenesených dat.

V Altus Variu se po instalaci před započítím veškerých prací s daty definují knihy agend, podle kterých se data budou odlišovat. Pro agendu Adresář, která obsahuje údaje organizací a osob budu ukládat záznamy do knihy Adresář Eshop abych tak rozlišil tyto kontakty od ostatních a mohl obnovovat pouze tyhle z téhle knihy.

Položky, které už existují a byly už pouze pozměny lze odlišit od těch nepozměněných pomocí data aktualizace a tak budu obnovovat pouze ty, které se liší právě tímto datem.

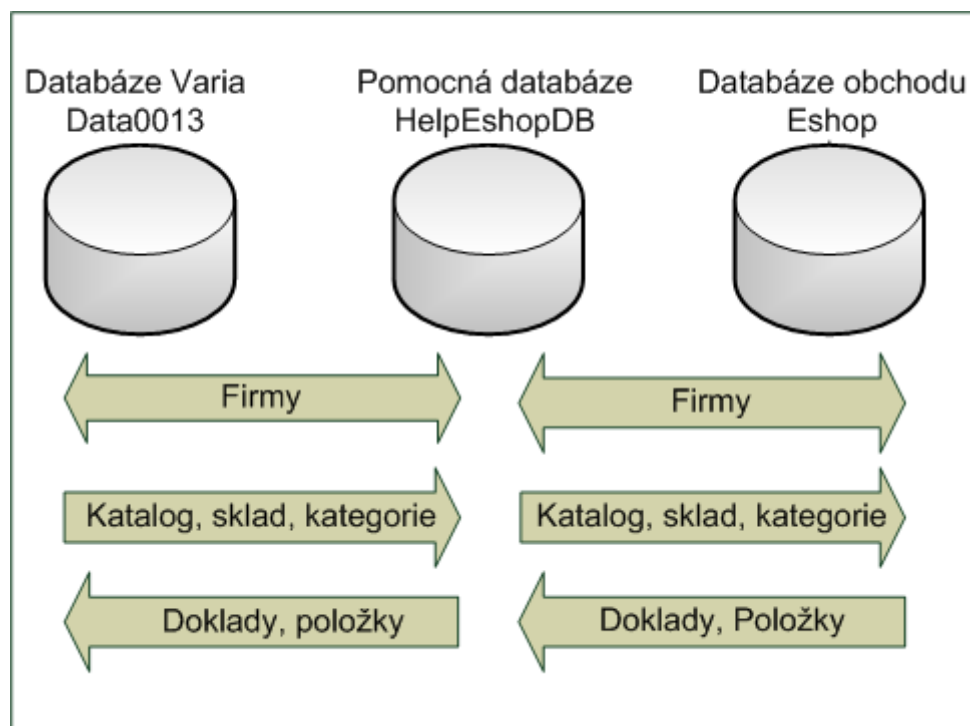
4.5 Postup při přenášení dat

Je velmi důležité dodržovat postup, které tabulky se obnoví nejdříve a které až naposled, jelikož tabulky obsahují cizí klíče ostatních tabulek, tzv. vazby. Například je tomu tak při ukládání objednávky s neregistrovaným zákazníkem. V momentě zaslání objednávky se nejdříve do databáze musí uložit zákazník, až pak doklad, který obsahuje vazbu na zákazníka, který si objednal zboží a jako poslední položky dokladu, které obsahují vazbu na doklad a také i na katalog.

4.6 Směry toku dat

Logicky jsou dva směry toku dat, kterými budou proudit data:

- Z Varia do pomocné a z pomocné do databáze internetového obchodu: Ceník, Firmy, Související produkty, Katalog, Kategorie, Kategorie produkty, Sklad. Tyto položky se obnovují buď v určeném časovém intervalu, nebo po interakci pracovníka.
- Z internetového obchodu do Varia: Firmy, Doklady, Položky dokladů. Tyto položky jsou aktualizovány okamžitě po vytvoření nebo po změně údajů.



Obrázek 4.1: Směry toku dat

4.7 Zajištění obnovy dat v určitém intervalu

MS SQL server 2008 R2 nemá možnost využívat JOBS funkci. Této funkci se nastavuje časový interval a přikáže se jí, co má udělat. Může to být cokoli, co zahrnuje práce v SQL serveru jako volání procedurálních funkcí, záloha dat atd.

Další možností je vytvoření Timer instance, což je časovač, kterému se nastaví čas. Po uplynutí času se zavolá funkce, ve které se provede právě synchronizace dat z Varia. Rozhodl jsem se tedy pro tuhle variantu.

```

Timer _timer = new Timer();
_timer.Interval = 60000; // some interval
_timer.Elapsed += new ElapsedEventHandler(_timer_Elapsed);
_timer.Start();
  
```

Timer je nastaven na interval 60000 milisekund a zavolá se odkazována funkce `_timer_Elapsed`, která obnoví data z Varia do internetového obchodu. Instance třídy Timer se spouští z `global.asax`, což je instance, která se naběhne jako první od spuštění e-shopu.

4.8 Co když server Varia v době odeslání objednávky a registrace nebude dostupný?

Internetový obchod by měl nezávisle na databázi Varia pracovat samostatně na své databázi. Pokud zrovna není přístupná databáze Varia, pak je třeba nějak nově vytvořené či změněné objednávky a registrace uchovat v databázi obchodu a pokusit se o přenos do Varia později.

Vyřešil jsem to tak, že každý řádek z tabulek Firmy, Doklady a PoložkyDokladu má příznak ZapsanoDoVaria, který je defaultně nastavený na false. Tudíž není zapsán ve Variu. Teprve až se úspěšně položka přenesla do Varia, nastaví se na true.

Pokud tomu tak ale není, postarají se o nápravu procedury KontrolaZákazníku, KontrolaDokladu a KontrolaPolozek dokladu, které pomocí cyklu projíždí tabulky a po objevení záznamů s nastaveným příznakem ZapsanoDoVaria se jej pokusí znovu odeslat.

Tyto procedury jsou volány po uplynutí časového intervalu spolu s obnovením dat z Varia do obchodu, nebo pomocí tlačítka Obnovit vše v admin sekci (login admin, heslo a jako klasický zákazník).

4.9 Online pohledy

Když se zákazník zaregistruje nebo vytvoří objednávku, jsou tato data okamžitě vidět i ve Variu. Ale jak je tomu když se data změní ve Variu ať údaje objednávky? Dejme tomu že zákazník chce změnit množství objednaných produktů. Telefonicky to oznámí obsluze obchodu, která změnu okamžitě provede. Už ne jednou jsem měl podobný problém a očekával jsem, že změna objednávky se projeví na webu okamžitě a ne až po nějaké hodině, kdy se pravděpodobně v časovém intervalu přenesly změněná data do internetového obchodu.

Ihned aktuální zjišťování informací jsem vyřešil pomocí tzv. online pohledů. Jsou to procedury (OnlinePohledDoklady, OnlinePohledPoložkyDokladu) umístěné v databázi internetového obchodu, které se dotazují do databáze Varia a vrací aktuální informace objednávek netrpělivému klientu. Pokud zrovna databáze Varia nebude přístupná tak se klient nedozví nic, ale to by se nemělo stávat, jelikož servery by měly ideálně běžet 100% času bez poruch.

5 Příprava jednoduché webové aplikace se základními funkcemi

Pro testování datového přenosu jsem vytvořil webovou aplikaci se základní funkcionalitou internetové obchodování.

5.1 Použitý software a technologie

Pro tuto aplikaci jsem rozhodl použít technologii ASP .NET, která je součástí Microsoft .Framework. Code-Behind stránek jsem realizoval v jazuku C#. Jako rozhraní pro komunikaci s SQL serverem jsem využil nástroje Linq to SQL, který je určen pro manipulaci s SQL daty. Protože tato aplikace poběží na jednom počítači jako SQL server databázi HelpEshopDB, využil jsem tohoto SQL serveru i pro databázi aplikace. Jako vývojové prostředí pro webovou aplikaci jsem zvolil Microsoft Visual Studio 2010, jehož součástí je i webový server pro ladění a testování.

5.2 Popis webové aplikace

Webová aplikace obchodu obsahuje pouze základní funkce, které dostatečně prověří propojení na ekonomický systém Altus Vario 12.

Levé menu kategorií se ukládá do cache, aby to při proklikávání zbytečně nezatěžovalo databázi obchodu. Po obnově dat z Varia se obnoví i menu a zobrazí se jen ty kategorie, které obsahují nějaký produkt.

Synchronizace dat se provádí v administrační sekci a to po přihlášení pomocí přihlašovacích údajů (login “admin” a heslo “a”) kliknutím na tlačítko Obnovit vše v sekci Menu. V tom okamžiku se přenesou objednávky a registrace do Varia, které byly vytvořeny po dobu nefunkčnosti Varia SQL serveru a z Varia do pomocné a z pomocné do databáze obchodu se přenesou a obnoví data.

Ve Variu u produktu v katalogu je možnost zatrhnout políčko Nabízet. Pokud bude odškrtnuté, produkt se nebude nabízet v e-shopu.

Výčet funkcí e-shopu:

- Registrace, přihlášení, osobní údaje, přehled objednávek
- Vytvoření objednávky (za zmínku stojí, že pokud neregistrovaný uživatel vytvoří objednávku, vytvoří se i nový záznam v tabulce Firmy, jelikož doklad se váže na Firmu)
- Obnova dat jako admin
- Přehled zboží

6 Závěr

Cílem této bakalářské práce byl návrh a implementace systému, který by datově propojil ekonomický systém Altus Vario s internetovým obchodem. Program Altus Vario již obsahuje funkce pro obchodování pomocí internetu. Všechny byly prověřené a bylo snahou některý z nich využít pro tento projekt. Prvně jsem doufal v podporu XML a využití exportu a importu XML dokumentů určené pro e-shop FastCentrik, ale po bližším prozkoumání se objevily požadavky, které by nedělaly aplikaci tak automatizovanou jak jsem původně zamýšlel. Mým největším favoritem bylo využití webových služeb AVIS, které byly v době, kdy jsem je testoval ještě ve vývoji. Kdybych si měl vybrat řešení zrovna teď, určitě bych si je zvolil.

Proto byl zvolen přímý přístup k databázi programu a jako prostředník mezi internetovým obchodem server MS SQL. Funkčnost řešení byla ověřena jednoduchou webovou aplikací, jejíž návrh a realizace byla také předmětem této práce.

Použitá literatura

- [1] Altus Vario: ekonomický software [online]. [2008] [cit. 2013-05-01]. Dostupné z: <http://www.vario.cz/>
- [2] Microsoft: Microsoft SQL Server [online]. © 2013 [cit. 2013-05-01]. Dostupné z: <http://msdn.microsoft.com/en-us/library/bb545450.aspx>
- [3] Microsoft: ASP.NET and Visual Studio [online]. © 2013 [cit. 2013-05-01]. Dostupné z: <http://msdn.microsoft.com/en-us/library/dd566231.aspx>

Seznam příloh

Součástí BP/DP je CD/DVD.

Adresářová struktura přiloženého CD/DVD:

Adresář	Obsah složky
Dokumentace	Datový slovník databáze Eshopu, návod k instalaci
SQLScripts	Zdrojové kódy pro SQL server
Eshop	Zdrojové kódy aplikace internetového obchodu
BackUpDB	Zaloha databáze HelpEshopDB a Eshop
VirtualBox	Disk s instalovaným systémem k odzkoušení